Eurographics Workshop on 3D Object Retrieval (2009)
I. Pratikakis, M. Spagnuolo, T. Theoharis, and R. Veltkamp (Editors)

SHREC 2009 - Shape Retrieval Contest of Partial 3D Models

H. Dutagaci[1], A. Godil[1], A. Axenopoulos[2], P. Daras[2], T. Furuya[3], R. Ohbuchi[3]

[1]National Institute of Standards and Technology, USA
[2]Informatics and Telematics Institute, Centre for Research and Technology Hellas, Thessaloniki, Greece
[3]Graduate School of Medical and Engineering Science, University of Yamanashi, Japan

Abstract

The objective of the Shape Retrieval Contest '09 (SHREC'09) of Partial Models is to compare the performances of algorithms that accept a range image as the query and retrieve relevant 3D models from a database. The use of a range scan of an object as the query addresses real life scenarios, where the task of the system is to analyze a 3D scene and to identify what type of objects are present in the scene. Another benefit of developing retrieval algorithms based on range scans of objects is that they enable a simple 3D search interface composed of a desktop 3D scanner. Two groups have participated in the contest and have provided rank lists for the query set that is composed of range scans of 20 objects. This paper presents descriptions of the participants' methods and the results of the contest.

Categories and Subject Descriptors (according to ACM CCS): I.4.8 [Image Processing and Computer Vision]: Scene Analysis, H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

1. Introduction

3D object retrieval is a relatively new field that pose many challenges. A major effort of the research community has been devoted to the formulation of accurate and efficient 3D object retrieval algorithms. In most of the existing state-of-the-art approaches, a complete 3D model has to be provided as a query in order to retrieve similar results. However, in real life scenarios a complete 3D model may not be always available. The input may be a partial model or a range scan of an object.

The major benefit of a retrieval system that can handle partial models, specifically range scans, is its applicability to robot vision. In robotics, the ability to process range images is critical for a robot to analyze its environment, navigate through the environment, and handle the objects of interest. In addition to robotics, there are other applications such as automated inspection, satellite image analysis, and automatic target detection and recognition. 3D face recognition from range scans is, by itself, a major research topic. In this respect, research on range image recognition is older than the 3D model retrieval area [BJ85].

However in most of the studies on range image recognition, the input scene is assumed to contain one of the target objects in the database and the objective is to recognize that particular object. The surface structure is only modified by the acquisition noise. The problem is to match the input surface to the corresponding model, which is already available to the system. In [RCSM03], Ruiz-Correa et al. point to this issue and discuss the alignment-verification tradition in range image analysis.

The retrieval problem is more general than identifying an object. First of all, the query view does not necessarily belong to any of the target models in the database. The task is to determine what kind of object is present in the scene rather then which particular object. The system should assess the relevance between the view of an unseen object and the models in the database. In addition to acquisition noise and the lack of pose and scale information, the system

Disclaimer: Any mention of commercial products or reference to commercial organizations is for information only; it does not imply recommendation or endorsement by NIST nor does it imply that the products mentioned are necessarily the best available for the purpose.

should also be able to deal with large intra-class variations. The challenges of the multimedia retrieval systems, such as the semantic gap, the difficulty of defining categories, the dependence on user preferences and the organization of the target database, also apply here. Therefore, the range queries should be processed with regard to the current issues raised by the 3D object retrieval community [TV07].

Another benefit of retrieval algorithms operating on range images is that they enable a 3D model search interface based on a desktop 3D scanner. The user has a small physical object of interest and wants to retrieve similar objects from a 3D repository. A simple desktop laser scanner acquires the range image of the object from an arbitrary view direction and the scan is introduced to the system as the query.

One category of 3D model retrieval algorithms is the view-based approach, which is inherently suitable for the range-based query input. In general, a number of views (depth maps or silhouettes) of the 3D object are extracted and encoded with feature vectors. The similarity of two complete 3D models is assessed in terms of the distance between the feature vectors of their corresponding views. This approach should be modified such that the input range image is compared to the views of the target models in an efficient manner. The two participants of the SHREC'09 Shape Retrieval Contest of Partial Models have provided such view-based algorithms and we believe that their contribution will lead to application of other view-based approaches to this specific problem.

2. The Data Set

2.1. Target Set

The target database is a subset of the shape benchmark constructed at NIST, desribed in [FGLW08]. It contains 720 complete 3D models, which are categorized into 40 classes. The classes are defined with respect to their semantic categories and are listed in Table 1. In each class there are 18 models. The file format to represent the 3D models is the ASCII Object File Format (*.off).

Bird	Fish	NonFlyingInsect
FlyingInsect	Biped	Quadruped
ApartmentHouse	Skyscraper	SingleHouse
Bottle	Cup	Glasses
HandGun	SubmachineGun	MusicalInstrument
Mug	FloorLamp	DeskLamp
Sword	Cellphone	DeskPhone
Monitor	Bed	NonWheelChair
WheelChair	Sofa	RectangleTable
RoundTable	Bookshelf	HomePlant
Tree	Biplane	Helicopter
Monoplane	Rocket	Ship
Motorcycle	Car	MilitaryVehicle
Bicycle		

Table 1: *40 classes of the target database.*

Figure 1: *3D partial models.*

2.2. Query Set

In the Partial Model Retrieval track, there are two different query sets. The first query set consists of 20 3D partial models which are obtained by cutting parts from complete models (Figure 1). The objective is to retrieve the models which have parts similar to the query. However, we did not receive rank lists for this first query set, hence we exclude the discussion on parts-based query from this paper. Interested readers may refer to the track website [SHR] to download the partial query set.

The second query set is composed of 20 range images, which are obtained by capturing range data of 20 objects from arbitrary view directions. Figure 2 and Figure 3 show the test objects and their range scans, respectively. The range images were captured using the NextEngine desktop 3D scanner. This kind of small device is suitable for a user interface because the user does not have to submit a full 3D computer model or a 2D sketch but the scan of an actual object. Figure 4 shows the setup for the acquisition of the range images. The range scans are converted to a triangular mesh and are saved in the ASCII Object File Format (*.off).

As can be observed from Figure 3, the range scans have imperfections as opposed to depth maps that are artificially generated from complete 3D models. The surfaces include holes and unconnected regions. Some geometric information is missing not because of the self-occlusion but because of the limited range of the 3D scanner. Furthermore, the re-

flectance properties of the object's surface greatly affects the quality of the scan. Non-smooth surfaces cause scattering of the laser light. The light is not reflected properly from dark regions or regions under shadow. These factors prevent the laser scanner from accurately reading the geometric information of those regions.

Figure 2: *The objects scanned to obtain the query views.*

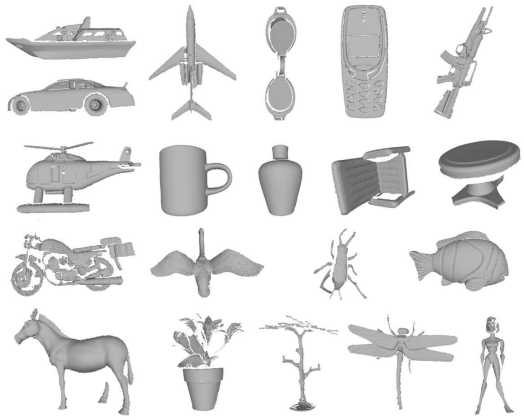


Figure 3: *The query views.*

3. Evaluation Measures

The participants have submitted rank lists for the query inputs. The length of each rank list is equal to the size of the target database. Using the rank lists the following evaluation measures were calculated: 1) Nearest Neighbor (NN), 2) First Tier (FT), 3) Second Tier (ST), 4) E-measure (E), and 5) Discounted Cumulative Gain (DCG). Details regarding the calculation and significance of these measures can be found in [SMKF04]. In addition to these scalar performance measures, the precision-recall curves were also obtained.

4. Submissions

Two groups have participated in the SHREC'09 Shape Retrieval Contest of Partial Models. A. Axenopoulos and P. Daras from Centre for Research and Technology Hellas, Thessaloniki have participated with three methods based on their Compact Multi-View Descriptor (CMVD) approach. The CMVD-Binary method uses only the silhouettes of 3D objects, whereas the CMVD-Depth method processes the depth maps. Their third descriptor, which we will refer to as "Merged", corresponds to the fusion of the CMVD-Binary and CMVD-Depth methods. The Compact Multi-View Descriptor is explained in Section 5.

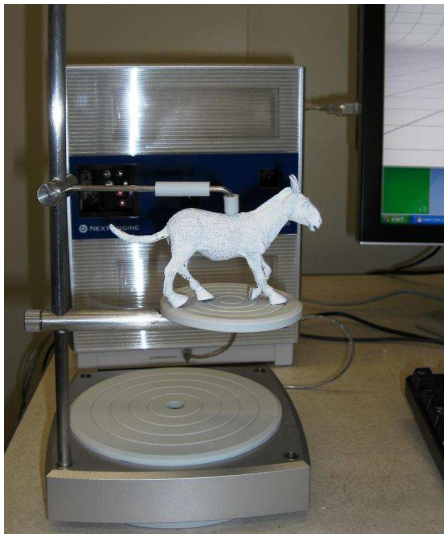


Figure 4: *The setup for the acquisition of the range images.*

T. Furuya and R. Obhuchi from University of Yamanashi have submitted two sets of rank lists corresponding to the BF-SIFT and BF-GridSIFT methods. These two methods are based on the ideas of Bag of Features (BF) and the Scale Invariant Feature Transform (SIFT). A brief description of the method is given in Section 6. The reader may refer to the paper of Obhuchi et al. [OOFB08] for a detailed discussion about the approach.

5. Compact Multi-View Descriptor (CMVD)

Compact multi-view descriptor is a novel view-based approach for 3D object retrieval. A set of 2D images (multi-views) are automatically generated from a 3D object, by taking views from uniformly distributed viewpoints. For each image, a set of 2D rotation-invariant shape descriptors is extracted. The Compact Multi-View Descriptor (CMVD) belongs to the category of the 2D view-based approaches, and thus, has the advantage of being highly discriminative, can be effective for partial matching and can support a variety of queries, such as 2D images, hand-drawn sketches and 3D range scans.

5.1. Descriptor extraction method

For the 3D objects stored in a database, the descriptor extraction procedure can be summarized in the block diagram presented in Figure 5. The input 3D object is a triangulated mesh, in one of the common 3D file formats (VRML, OFF, 3DS, etc.). As a first step, a pose normalization takes place, which includes translation, scaling and rotation of the object. After the pre-processing step, a set of 18 2-dimensional views, taken from the vertices of a bounding 32-hedron is extracted. Both binary (black/white) and depth images are generated. To each of the extracted 2D images, a set of 2D functionals is applied, resulting in a descriptor vector for each view.

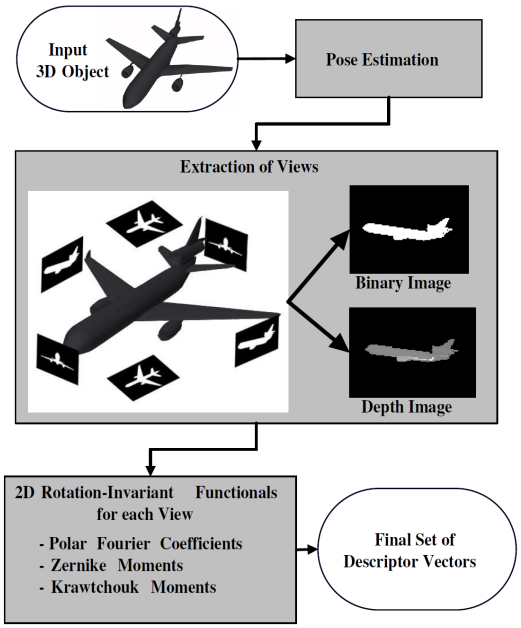


Figure 5: *Block Diagram of the extraction of the Compact Multi-View Descriptor.*

5.2. A set of uniformly distributed views

The CMVD is based on the matching of multiple 2D views, which can be extracted from a 3D object by selecting a set of different viewpoints. In order to be uniformly distributed, the viewpoints are chosen to lie at the vertices of a regular polyhedron. The type of the polyhedron and the level of tessellation need to be carefully considered in order to provide the optimal solution. As mentioned in [CTSO03], 15 to 20 views can roughly represent the shape of a 3D model. Based on this notion, the 18 vertices of the 32-hedron, which are produced by tessellation of octahedron at the first level, can provide an appropriate set of viewpoints.

In order to render the multi-view images, the camera viewpoints are placed at the 18 vertices of the 32-hedron. Two 2D image types are available: 1) Binary Images: The rendered images are only silhouettes, where the pixel values are 1 if the pixel lies inside the model's 2D view and 0 otherwise. 2) Depth Images: The pixel intensities are proportional to the distance of the 3D object from each sample point of the corresponding tangential plane.

Although binary images provide an efficient and robust representation of a 2D view, depth images contain more information and produce better retrieval results, if appropriately exploited.

5.3. Computing 2D descriptors on each view

The set of uniformly distributed views, described in 5.2, consists of 2D binary images and depth images of size 100×100 pixels. To each image, three rotation-invariant descriptors [ZDA*07] are applied in order to produce the final set of descriptors per view.

Let $f_t(i,j)$ be the 2D image, where $i,j = 0,...,N-1$ and $N \times N$ is the size of the image, $t = 1,...,N_V$ and N_V is the total number of views. The values of $f_t(i,j)$ are either 0 or 1 for the binary images, while in the case of depth images, the values can be any real number between 0 and 1.

2D Polar-Fourier Transform. The Discrete Fourier Transform (DFT) is computed for each $f_t(i,j)$, producing the vectors $FT(k,m)$, where $k,m = 0,...,N-1$. In the DFT, shifts in the spatial domain correspond to linear shifts in the phase component. Thus, the DFT magnitude is invariant to circular translation. Therefore, using discrete polar coordinates, rotation is converted to circular translation, which leads to rotation-invariant descriptors. For each $f_t(i,j)$, the first $K \times M$ harmonic amplitudes are considered.

2D Zernike Moments. Zernike moments are defined over a set of complex polynomials which form a complete orthogonal set over the unit disk and are rotation invariant. The Zernike moments Z_{km} [PVMRGA04], where $k \in N^+$, $|m| \leq k$, are calculated for each $f_t(i,j)$ with spatial dimension $N \times N$, produce a vector of rotation-invariant Zernike descriptors.

2D Krawtchouk Moments. Krawtchouk moments are a set of moments formed by using Krawtchouk polynomials as the basis function set. Following the analysis in [YRO03] and some specifications mentioned in [Tea80], they were computed for each $f_t(i,j)$, producing a vector of rotation invariant Krawtchouk descriptors.

A compact representation of the multi-view descriptor implies a small number of descriptors per view, otherwise the shape matching time would be prohibitive. An optimal number of descriptors N_D for each view, which was found experimentally, is given below:

$$N_D = N_{FT} + N_{Zern} + N_{Kraw} \quad (1)$$

where $N_{FT} = 78$, $N_{Zern} = 56$ and $N_{Kraw} = 78$. Finally, two types of descriptors are formed: CMVD-Binary that uses binary images and CMVD-Depth that uses depth images.

A similar procedure is followed for descriptor extraction in 3D range scans. In this case, a single view is extracted from the scanned image, which is also given as a triangulated mesh. Both binary and depth images are extracted from this view.

5.4. Matching Method

Retrieval of 3D models can be achieved if, instead of a 3D model, a single range image is used as a query. In order to measure the dissimilarity, the query range image is compared to the N_V views of the 3D model and the most similar (to the image) view is selected(Figure 6).

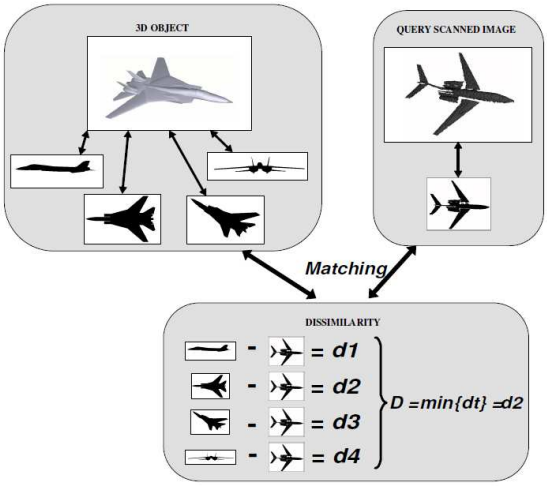


Figure 6: *Similarity Matching Framework for the Compact Multi-View Descriptor.*

Let $\mathbf{D}_t$ be the descriptor vector of the t^{th} view, which is extracted according to the procedure described in 5.3. The dissimilarity metric between the t^{th} view of a 3D object A and a query range image Q is given by the L1-distance:

$$d_t = \sum_{k=1}^{N_D} |D_t^A(k) - D^Q(k)| \qquad (2)$$

where N_D is the number of descriptors per view. Finally, the view that produces the lowest dissimilarity to the query image is selected:

$$d = \min\{d_t\} = \min\{\sum_{k=1}^{N_D} |D_t^A(k) - D^Q(k)|\} \qquad (3)$$

where $t = 1,...,N_V$, $N_V = 18$ is the total number of views of model A, $D^Q(k)$ are the descriptors of the query image Q and $D_t^A(k)$ are the descriptors of the t^th view of model A.

It is obvious that 2D-3D matching cannot be as efficient as 3D-3D matching, since a 2D image is unable to capture the global geometric information of an object. However, it is much easier to provide a 2D image as query than a 3D model, either by taking a photo or by using a range scanner and acquiring a depth image.

In Table 2, the average computation times for descriptor extraction and matching procedures are summarized. The times were obtained using a PC with a 2.4 GHz processor and 3GB RAM, running the operating system Windows XP.

Action	Time (msec)
Views Generation	2587
Polar-Fourier Descriptors Extraction	63
Krawtchouk Descriptors Extraction	398
Zernike Descriptors Extraction	811
Matching between 2 views	0.4

Table 2: *Average computation times for descriptor extraction and matching procedures of the CMVD approach.*

6. Bag of Features - Scale Invariant Feature Transform (BF-SIFT)

In this section the BF-SIFT approach, used in the competition by University of Yamanashi, is described. The readers can refer to [OOFB08] for a more detailed description. The system compares shapes of 3D models visually by using a set of local features extracted from multiple view 2D depth images of the model. As the method employs so called bag-of-features (BF) approach [CDF*04, WCM05, SZ03] to integrate thousands of local visual features into a feature vector per 3D model, the algorithm is named as BF-SIFT.

The BF-SIFT algorithm compares 3D models by following the steps below:

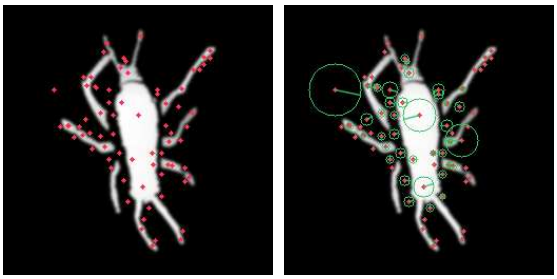
1. **Pose normalization (position and scale):** The BF-SIFT performs pose normalization, not only for position and scale, but also for rotation, so that the model is rendered with an appropriate size at the center of the view sphere in each of the multiple-view images.
2. **Multi-view rendering:** The system renders range images of the model from N_i viewpoints placed uniformly on the view sphere surrounding the model.
3. **SIFT feature extraction:** From the range images, it extracts local, multi-scale, multi-orientation, visual features by using the SIFT [Low04] algorithm. Two different feature extraction methods are available: The original SIFT and the GridSIFT.
4. **Vector quantization:** The system vector-quantizes a local feature into a visual word in a vocabulary of size N_v by using a visual codebook. The vector quantization is a nearest point search in a high dimensional space, and

the process is accelerated by a kd-tree. Prior to the retrieval, the visual codebook is learned, using an unsupervised learning approach based on (tens of) thousands of training features extracted from a set of training models, e.g., the models in the database to be retrieved. The learning is done by the well-known k-means clustering algorithm.

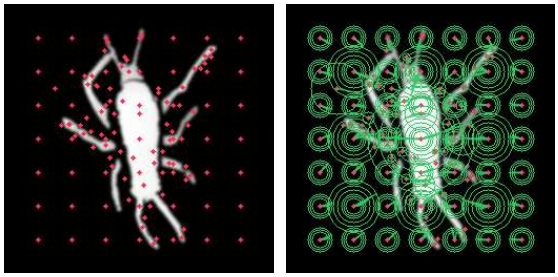
5. **Histogram generation:** Quantized local features or "visual words" are accumulated into a histogram having N_v bins. The histogram becomes the feature vector of the corresponding 3D model.
6. **Distance computation:** The algorithm compares the histogram generated from the query, a range image, with $N_i = 42$ histograms of a model in the database. The minimum of 42 distances among a query (a range image) and a 3D model (42 range images) becomes the distance between the query and the 3D model. That is, each distance between a pair of feature vectors (the histograms) is computed by using Kullback-Leibler Divergence (KLD):

$$D(x,y) = \sum_{i=1}^{n} (y_i - x_i) \ln \frac{y_i}{x_i} \quad (4)$$

where $\mathbf{x} = (x_i)$ and $\mathbf{y} = (y_i)$ are the feature vectors and n is the dimension of the vectors.



(a) Original SIFT interest points. (b) Original SIFT features.



(c) Grid-SIFT interest points. (d) Grid-SIFT features.

Figure 7: *Interest points and SIFT features for the Original SIFT and the Grid SIFT algorithms.*

Some modifications are applied to the SIFT feature extraction step. The SIFT features are sensitive to rapid intensity change at various scales. To remove spurious interest points, a simple anti-aliasing is employed for the range image rendering. The range images are first rendered at 1024×1024 resolution, filtered with low-pass Gaussian filter and then downsampled to 256×256.

The experiments are conducted with two different sampling patterns. The original SIFT algorithm first detects interest points by searching, in scale space, points of maximum response of a local, orientation sensitive gradient filter. Then, at each of these interest points, extracts a 128D SIFT feature. However, the interest point detector in the original SIFT algorithm may not be optimal for the task of the partial retrieval track. Figure 7(a) and Figure 7(b) shows examples of the interest points and features computed by using the (original) SIFT with interest point detector. For example, note that there are many interest points appeared near the gaps between legs and the body of this insect model. These gaps are artifact of range scanning (tangential planes, occlusions, etc.). The global shape would matter more than the "artificial" features at the gaps.

As a quick fix, interest points are added on a regular grid and the BF-GridSIFT variation of the BF-SIFT algorithm is obtained. A 7×7 grid for 49 feature points per range image is added. To capture global, lower frequency features by using the grid sampling, the grid sampling is applied only for the larger scales of the SIFT features. For the smaller scales (higher frequency bands), original interest points generated by the original SIFT algorithm are used as the sample points. Figure 7(c) and Figure 7(d) shows the interest points and the features of the BF-Grid-SIFT note that; (1) there are more sample points overall, (2) there are more larger scale features capturing global shape, and (3) proportion of samples at the gaps are smaller. Table 3 compares average numbers of sample points per 3D model of the (original) BF-SIFT and BF-GridSIFT approaches. BF-GridSIFT produced many more feature points. This produces on average 200 features per view, still not enough to construct a robust histogram. The retrieval performance can be increased if more SIFT features are sampled per range image so that the histogram is no exceedingly sparse.

	Number of features per model
BF-SIFT	1,131
BF-GridSIFT	8,222

Table 3: *Number of samples for the BF-SIFT and BF-GridSIFT.*

The vocabulary size is set to $N_v = 30$ for the BF-SIFT and $N_v = 800$ for the BF-GridSIFT, after some experiments. These vocabulary sizes are much smaller than those optimal for comparing 3D models (e.g., $Nv \sim 1200$). Having only a depth image to extract features from for the query, the number of meaningful features may have been limited.

Distance computation is also different from the original BF-SIFT [OOFB08]. For this track, the query is a range image from single viewpoint, while the database is a set of

complete 3D models. Thus, the distance computation needs a modification with respect to the original BF-SIFT algorithm. The original BF-SIFT computed a histogram per 3D model by bagging all the local features from N_i views. For the partial matching track, for each 3D model, N_i histograms are computed. For the query scan, only one histogram is available. Then, the distances between the query's histogram and a set of N_i histograms of the 3D model is computed and the minimum of the N_i distances is selected as the final dissimilarity measure. Distance among each pair of histograms is computed by using the Kullback-Leibler divergence.

7. Results

The participants of the SHREC'09 Shape Retrieval Contest of Partial Models submitted five sets of rank lists each corresponding to a different method. The results for the five methods are summarized in Table 4. The two best performing methods are BF-GridSIFT algorithm by Furuya and Ohbuchi, and the CMVD-Depth algorithm by Axenopoulos and Daras. Both of the two methods give the same nearest neighbor accuracy. The BF-GridSIFT algorithm gives slightly better results than the CMVD-Depth approach in terms of other measures, i.e. first tier, second tier, E-measure and discounted cumulative gain.

Another observation is that the merging of the CMVD-Depth and CMVD-Binary approaches results in lower performance values than the CMVD-Depth approach alone, except with the DCG. The gain in the DCG, however, is not significant. The depth map inherently contains the shape information of the binary silhouette and brings more discriminative features.

Method	NN	FT	ST	E	DCG
CMVD-Binary	0.350	0.217	0.283	0.200	0.521
CMVD-Depth	0.450	0.197	0.267	0.174	0.511
Merged	0.350	0.211	0.281	0.192	0.526
BF-SIFT	0.150	0.114	0.186	0.116	0.423
BF-GridSIFT	0.450	0.225	0.297	0.204	0.532

Table 4: *Average Retrieval Results over the 20 Query Scans.*

Figure 8 shows the precision-recall curves. The results for all three runs of the CMVD approach outperform the BF-SIFT method, while they are competitive with the BF-GridSIFT method. More specifically, the CMVD method outperforms BF-GridSIFT for recall values close to 0.1, from 0.5 to 0.6 and greater than 0.9. However the BF-GridSIFT method gives better precision values for the recall values between 0.2 and 0.5.

Figure 9 shows the individual DCG values of the two best performing methods for the 20 query objects together with their class identities. The BF-GridSIFT method gives higher DCG values for the first 10 objects and the CMVD-Depth method is better for the rest. For most of the objects the two

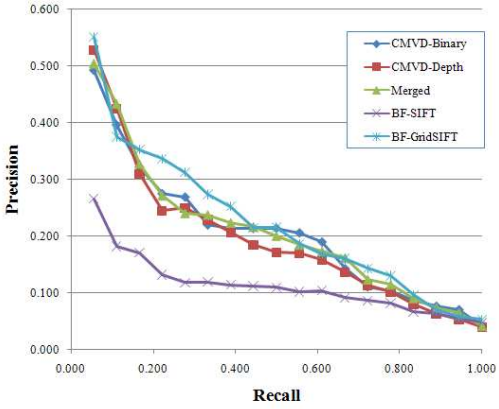


Figure 8: *Precision-recall curves. The ideal case would be a straight line at the top, with precision values at one for all the recall values. The closer the curve to this ideal case, the better the algorihtm is.*

algorithms achieve close performance. Both methods give good results for the objects "Monoplane", "Quadruped" and "Motorcycle", however there is significant performance difference for the objects "Glasses", "SubmachineGun", "Bird" and "Homeplant".

The web-based interface of the Partial Models Retrieval track shows the retrieved models for all the query objects and the five methods [INT]. We reproduce a sample shot from the interface in Figure 10.

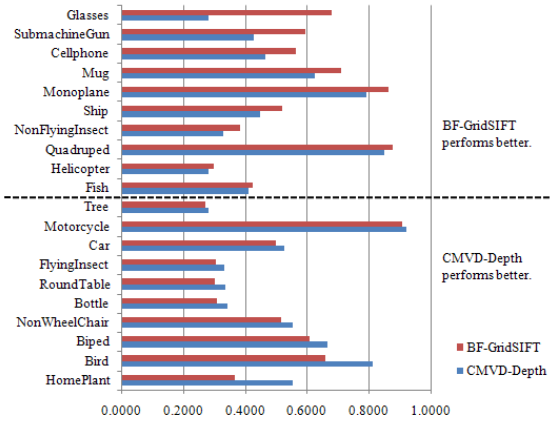


Figure 9: *DCG values with respect to the query views. For the first 10 objects BF-GridSIFT method gives higher DCG values whereas for the last 10 objects CMVD-Depth method performs better.*

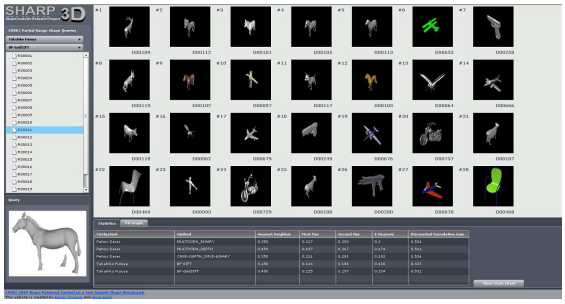


Figure 10: *A sample shot from the web-based interface of the SHREC'09 Shape Retrieval Contest of Partial Models [INT].*

8. Conclusions

In this paper, we have described and compared five algorithms of two research groups that participated in the SHREC'09 - Shape Retrieval Contest of Partial 3D Models. The algorithms accept a range scan as the input and retrieve similar models from a database of complete 3D models. The CMVD-depth and BF-GridSIFT methods yielded the best performance among the five algorithms. The two methods are complementary in the sense that one method retrieves more relevant models in response to some query objects, while the other method perform better for other query objects.

The results are encouraging, and we hope that this competition will lead to new research on range (or similar 2.5 input such as stereo) input-based 3D model retrieval. We are in progress of enlarging the range query set to create a new benchmark and we hope that it will provide a valuable contribution to the 3D Model retrieval, Robotic Vision and Perception Community.

9. Acknowledgements

The work of P. Daras and A. Axenopoulos was supported by the EC funded project VICTORY.

References

[BJ85] BESL P. J., JAIN R. C.: Three-dimensional object recognition. *ACM Comput. Surv. 17*, 1 (1985), 75–145.

[CDF*04] CSURKA G., DANCE C. R., FAN L., WILLAMOWSKI J., BRAY C.: Visual categorization with bags of keypoints. In *In Workshop on Statistical Learning in Computer Vision, ECCV* (2004), pp. 1–22.

[CTSO03] CHEN D.-Y., TIAN X.-P., SHEN Y.-T., OUHYOUNG M.: On visual similarity based 3D model retrieval. *Computer Graphics Forum 22*, 3 (Sept. 2003), 223–232.

[FGLW08] FANG R., GODIL A., LI X., WAGAN A.: A new shape benchmark for 3D object retrieval. In *ISVC (1)* (2008), pp. 381–392.

[INT] http://control.nist.gov/sharp/NSHREC/Partial-Range/SHREC .

[Low04] LOWE D. G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision 60*, 2 (2004), 91–110.

[OOFB08] OHBUCHI R., OSADA K., FURUYA T., BANNO T.: Salient local visual featuers for shape-based 3D model retrieval. In *Shape Modeling International* (2008).

[PVMRGA04] PADILLA-VIVANCO A., MARTINEZ-RAMIREZ A., GRANADOS-AGUSTIN F.-S.: Digital image reconstruction using zernike moments. In *Proceedings of the SPIE* (2004), vol. 5237, pp. 281–289.

[RCSM03] RUIZ-CORREA S., SHAPIRO L. G., MEILA M.: A new paradigm for recognizing 3-D object shapes from range data. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision* (2003), p. 1126.

[SHR] http://www.itl.nist.gov/iad/vug/sharp/benchmark/shrecPartial/ .

[SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The princeton shape benchmark. In *Shape Modeling International* (2004).

[SZ03] SIVIC J., ZISSERMAN A.: Video Google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision* (2003), vol. 2, pp. 1470–1477.

[Tea80] TEAGUE M. R.: Image analysis via the general theory of moments. *J. Opt. Soc. Am. 70*, 8 (1980), 920–930.

[TV07] TANGELDER J., VELTKAMP R.: A survey of content based 3d shape retrieval methods. *Multimedia Tools and Applications* (2007).

[WCM05] WINN J., CRIMINISI A., MINKA T.: Object categorization by learned universal visual dictionary. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision* (2005), pp. 1800–1807.

[YRO03] YAP P.-T., RAVEENDRAN P., ONG S.-H.: Image analysis by krawtchouk moments. *IEEE Transactions on Image Processing 12*, 11 (2003), 1367–1377.

[ZDA*07] ZARPALAS D., DARAS P., AXENOPOULOS A., TZOVARAS D., STRINTZIS M. G.: 3D model search and retrieval using the spherical trace transform. *EURASIP J. Appl. Signal Process. 2007*, 1 (2007).